View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Research Online

UNIVERSITY
OF WOLLONGONG
AUSTRALIA

University of Wollongong
Research Online

Faculty of Engineering and Information Sciences - Papers: Part A

Faculty of Engineering and Information Sciences

2013

Membership encryption and its applications

Fuchun Guo
University of Wollongong, fuchun@uow.edu.au

Yi Mu
University of Wollongong, ymu@uow.edu.au

Willy Susilo
University of Wollongong, wsusilo@uow.edu.au

Vijay Varadharajan
Macquarie University

Publication Details

Guo, F., Mu, Y., Susilo, W. and Varadharajan, V. (2013). Membership encryption and its applications. Lecture Notes in Computer Science, 7959 219-234.

Research Online is the open access institutional repository for the University of Wollongong. For further information contact the UOW Library: research-pubs@uow.edu.au

Membership encryption and its applications

Abstract
We propose a new encryption primitive called Membership Encryption. Let $P(G)$ be a privacy-preserving token on a group attribute/identity G, such that given $P(G)$ it is hard to know the attributes in G. In this membership encryption, if an encryption takes as input an attribute A and the token $P(G)$, the decryption requires holding the membership $A \in G$, i.e., A belongs to this group attribute. Membership encryption is applicable in constructing membership proof $A \in P(G)$ with privacy preserving on group attribute and the membership. Membership encryption can be also utilized to construct an efficient two-round K-out-of-N oblivious transfer protocol. In this paper, we construct a provably secure membership encryption where the group token $P(G)$ is constant-size with maximum number accountability on attributes. Using our scheme, the proposed oblivious transfer protocol exhibits the nice feature of $O(1)$ communication cost for any K from receiver to sender, and $O(N)$ communication cost from sender to receiver.

Keywords
applications, encryption, membership, its

Disciplines
Engineering | Science and Technology Studies

Publication Details
Guo, F., Mu, Y., Susilo, W. and Varadharajan, V. (2013). Membership encryption and its applications. Lecture Notes in Computer Science, 7959 219-234.

This journal article is available at Research Online: http://ro.uow.edu.au/eispapers/1323

Membership Encryption and Its Applications*

Fuchun Guo[1], Yi Mu[1], Willy Susilo**[1], and Vijay Varadharajan[2]

[1] Centre for Computer and Information Security Research
School of Computer Science and Software Engineering
University of Wollongong, Wollongong, Australia
{fg278,ymu,wsusilo}@uow.edu.au
[2] Information and Networked Systems Security Research
Department of Computing, Faculty of Science
Macquarie University, Sydney,Australia
vijay.varadharajan@mq.edu.au

Abstract. We propose a new encryption primitive called *Membership Encryption*. Let $\mathcal{P}(\mathbf{G})$ be a privacy-preserving token on a group attribute/identity $\mathbf{G}$, such that given $\mathcal{P}(\mathbf{G})$ it is hard to know the attributes in $\mathbf{G}$. In this membership encryption, if an encryption takes as input an attribute A and the token $\mathcal{P}(\mathbf{G})$, the decryption requires holding the membership $A \in \mathbf{G}$, i.e., A belongs to this group attribute. Membership encryption is applicable in constructing membership proof $A \in \mathcal{P}(\mathbf{G})$ with privacy preserving on group attribute and the membership. Membership encryption can be also utilized to construct an efficient two-round K-out-of-N oblivious transfer protocol. In this paper, we construct a provably secure membership encryption where the group token $\mathcal{P}(\mathbf{G})$ is constant-size with maximum number accountability on attributes. Using our scheme, the proposed oblivious transfer protocol exhibits the nice feature of $O(1)$ communication cost for any K from receiver to sender, and $O(N)$ communication cost from sender to receiver.

1 Introduction

Membership Proof. Proving that an attribute A belongs to a group attribute $\mathbf{G}$, denoted by group membership $A \in \mathbf{G}$, is useful and non-trivial especially when privacy protections are essential. Let $\mathcal{P}(O)$ denote privacy protection (e.g. commitment) on the object O, such that given $\mathcal{P}(O)$ it is hard to know the object O. The privacy-preserving membership proof falls into two different cases:

- $\mathcal{P}(A) \in \mathbf{G}$. The verifier knows the token $\mathcal{P}(A)$ and all attributes in $\mathbf{G}$. The prover wants to prove that the attribute in $\mathcal{P}(A)$ belongs to $\mathbf{G}$ without leaking the real attribute A to the verifier. Assuming that each attribute is an individual, this membership proof is towards privacy protection on the involved individual. We found the technique called set membership proof [11, 8, 6] is proposed for $\mathcal{P}(A) \in \mathbf{G}$.

* This work is supported by ARC Discovery Grant DP110101951
** W. Susilo is supported by ARC Future Fellowship FT0991397.

– $A \in \mathcal{P}(\mathbf{G})$. The verifier knows the attribute A and the token $\mathcal{P}(\mathbf{G})$. The prover wants to prove that the group attribute in $\mathcal{P}(\mathbf{G})$ contains A without leaking other attributes in $\mathbf{G}$ to the verifier. This membership proof is aiming at protecting the privacy of non-involved individuals. We found the technique called accumulator with witness $[3, 2, 17, 14, 7, 1]$ can be seen as a membership proof for $A \in \mathcal{P}(\mathbf{G})$.

Membership proof is useful in those privacy-preserving applications (see [11, 8, 6, 3, 2, 17, 14, 7, 1]), where $\mathcal{P}(O)$ instead of O is certified for privacy purpose, and the prover wants to prove that the certified $\mathcal{P}(O)$ satisfies some membership.

Motivation. In this work, we extend the membership by proof to membership by encryption. We are interested in exploring the notion of *membership encryption*. Let $\mathbf{G} = \{A_1, A_2, \cdots, A_k\}$ be a finite set of group attribute, and $\mathcal{P}(\mathbf{G})$ denote the privacy-preserving group $\mathbf{G}$. Following the membership proof $A \in \mathcal{P}(\mathbf{G})$, the membership encryption $A \in \mathcal{P}(\mathbf{G})$ is defined as follows: when the encryption takes as input A and $\mathcal{P}(\mathbf{G})$, the decryption requires holding the membership $A \in \mathbf{G}$.

We focus on the membership encryption $A \in \mathcal{P}(\mathbf{G})$ only as it can be naturally transferred into the membership encryption $\mathcal{P}(A) \in \mathbf{G}$, when $\mathcal{P}(\cdot)$ contains only one attribute. For example, let $\mathbf{G} = \{A_1, A_2\}$. To generate a membership encryption $\mathcal{P}(A) \in \{A_1, A_2\}$, we run membership encryption $A_1 \in \mathcal{P}(A)$ and $A_2 \in \mathcal{P}(A)$ on the same message and R, where $R = \{r_1, r_2\}$ and r_i is the randomness for $A_i \in \mathcal{P}(A)$ encryption. It is not hard to verify that this is equivalent to the membership encryption $\mathcal{P}(A) \in \mathbf{G}$.

Encryption vs Proof. Membership encryption is more powerful compared to membership proof in terms of three reasons. Firstly, a membership proof $A \in \mathcal{P}(\mathbf{G})$ cannot be converted into a membership encryption, but a successful decryption of membership encryption with A and $\mathcal{P}(\mathbf{G})$ as input naturally implies the membership $A \in \mathcal{P}(\mathbf{G})$. Secondly, given a membership proof, the verifier might be able to compromise the privacy of $\mathcal{P}(\mathbf{G})$ to others by publishing the membership proof $A \in \mathcal{P}(\mathbf{G})$. While the membership proof from membership encryption is non-transferable. Finally, considering the scenario that Alice would send a message to Bob if he can prove the membership $A \in \mathcal{P}(\mathbf{G})$. Using the membership proof, Bob needs to generate the proof first and then Alice sends messages to Bob after checking the proof, which costs two separated steps. Membership encryption combines the two steps into one, which improves the communication efficiency.

Membership encryption is also useful in other applications. One of them is the oblivious transfer protocol [19]. Suppose there are N messages $M_1, M_2, \cdots, M_N$, and a receiver wants to get part of them without leaking her/his choice to the message owner (sender). Using the membership encryption, the receiver generates $\mathcal{P}(C)$ and sends it to sender, where $C \subseteq \{1, 2, \cdots, N\}$ is the receiver's choice. The sender then encrypts message M_i with the index i and $\mathcal{P}(C)$. If $i \in C$, the receiver can decrypt the message M_i; otherwise $i \notin C$, the receiver will not be able to extract M_i. Suppose the number of choices is accountable

from $\mathcal{P}(C)$, we will obtain a two-round K-out-of-N protocol for any K from the membership encryption $A \in \mathcal{P}(\mathbf{G})$.

Contributions. We propose a new encryption primitive called *membership encryption* where the decryption satisfies the privacy-preserving group membership $A \in \mathcal{P}(\mathbf{G})$. To be precise, in our membership encryption definition, $\mathcal{P}(\mathbf{G})$ is generated from the group attribute $\mathbf{G}$ and a secret token $\mathbf{S}$. If the encryption takes as input A and $\mathcal{P}(\mathbf{G})$, the decryption is successful if and only if the decryptor knows $(\mathbf{G}, \mathbf{S})$ and $A \in \mathbf{G}$ is true.

We construct a provably secure membership encryption, which exhibits the following nice features.

- The group token $\mathcal{P}(\mathbf{G})$ is constant-size and independent of the number of attributes in $\mathbf{G}$.
- The upper bound attribute number in $\mathcal{P}(\mathbf{G})$ is accountable.
- The ciphertext is constant-size and dependent on the length of security parameter only.

We show how to apply membership encryption in constructing a two-round K-out-of-N oblivious transfer protocol $\mathsf{OT}^{\mathsf{K}}_{\mathsf{N}}$ for any $1 \leq K \leq N$. Our protocol satisfies the security model defined in [10]. In our protocol, messages from receiver to sender are the group token $\mathcal{P}(\mathbf{G})$ only and messages from sender to receiver are N constant-size ciphertexts. Using our proposed scheme, the communication cost from receiver to sender is $O(1)$ or constant-size, and communication cost from sender to receiver is $O(N)$ or linear in N. This is the first two-round $\mathsf{OT}^{\mathsf{K}}_{\mathsf{N}}$ protocol with the least communication cost compared to existing two-round oblivious transfer protocols [16, 18, 10, 9].

Roadmap. The rest of this paper is organized as follows. We give the definition and security models of membership encryption in Section 2. Our construction is proposed in Section 3 with the security proof in Section 4. We show how to apply membership encryption to the construction of two-round K-out-of-N oblivious transfer protocols in Section 5. In the final section, we conclude this paper.

2 Membership Encryption

2.1 Description of Membership Encryption

A membership encryption $A \in \mathcal{P}(\mathbf{G})$ with maximum number accountability on group attribute consists of the following five algorithms:

Setup: Taking as input a security parameter 1^λ, an integer n and all attributes $\{A_1, A_2, \cdots, A_n\}$, the setup algorithm generates the system parameter SP. Here, n denotes the upper bound attribute number of group tokens.

GroupGen: Taking as input the system parameter SP and a group attribute $\mathbf{G} = \{A_1, \cdots, A_k\}$ $(1 \leq k \leq n)$, the group token generation algorithm returns the token $\mathcal{P}(\mathbf{G})$ and the secret key $\mathbf{S}$.

Verify: Taking as input $\mathcal{P}(\mathbf{G})$ and an integer k, the verification algorithm returns true if the attribute number in $\mathcal{P}(\mathbf{G})$ satisfying $|\mathcal{P}(\mathbf{G})| \leq k$; otherwise, outputs false.

Encrypt: Taking as input the system parameter SP, an attribute A, a group token $\mathcal{P}(\mathbf{G})$ and a message M, the encryption algorithm returns a ciphertext C of M. We define the ciphertext as $C \leftarrow \mathsf{ME}[A, \mathcal{P}(\mathbf{G}), M]$.

Decrypt: Taking as input the attribute A, the group attribute $\mathbf{G}$, the secret key $\mathbf{S}$ and the ciphertext C, the decryption algorithm returns the message M or $\perp$. We define the decryption as $\{M, \perp\} \leftarrow \mathsf{MD}[C, \mathbf{G}, \mathbf{S}]$.

Correctness: The membership encryption must satisfy that for any system parameter SP, group token $(\mathcal{P}(\mathbf{G}), \mathbf{G}, \mathbf{S})$ and ciphertext $\mathsf{ME}[A, \mathcal{P}(\mathbf{G}), M]$, if $A \in \mathbf{G}$, we have $\mathsf{MD}[\mathsf{ME}[A, \mathcal{P}(\mathbf{G}), M], \mathbf{G}, \mathbf{S}] = M$; Otherwise, $A \notin \mathbf{G}$, we have $\mathsf{MD}[\mathsf{ME}[A, \mathcal{P}(\mathbf{G}), M], \mathbf{G}, \mathbf{S}] = \perp$.

2.2 Security Models of Membership Encryption

Definition 1 (Message Security) *A membership encryption captures the message security if given a ciphertext generated with A and $\mathcal{P}(\mathbf{G})$, it is computationally hard to know the encrypted message when*

- *The decryptor does not have the secret key $\mathbf{S}$ of $\mathcal{P}(\mathbf{G})$, or*
- *The attribute A does not satisfy the membership, i.e., $A \notin \mathbf{G}$.*

We define two games to capture message security. The first game is about *indistinguishability against secret key* and says that if the corresponding secret key $\mathbf{S}$ is unknown, it is indistinguishable to decide the message in a ciphertext for the corresponding token $\mathcal{P}(\mathbf{G})$ and any attribute A. The second game is about *indistinguishability against membership* and says that it is indistinguishable to decide the message in a ciphertext for any attribute A and any group token $\mathcal{P}(\mathbf{G})$ if $A \notin \mathbf{G}$ holds.

Game 1: Indistinguishability against Secret Key.

- **Setup:** The challenger runs the Setup algorithm to generate the system parameter SP, and sends it to the adversary.
- **Phase 1:** The adversary queries group tokens and decryption as follows.
  - For a token query on group attribute $\mathbf{G}_i$ that is adaptively chosen by the adversary, the challenger responds by generating $(\mathcal{P}(\mathbf{G}_i), \mathbf{S}_i)$ and sending $\mathcal{P}(\mathbf{G}_i)$ to the adversary.
  - For a decryption query on a ciphertext C_i for $(A, \mathcal{P}(\mathbf{G}_i))$ where $\mathcal{P}(\mathbf{G}_i)$ is generated by the challenger, if $A \notin \mathbf{G}$, the challenger returns $\perp$ to the adversary; otherwise, the challenger responds by decrypting the ciphertext with $\mathbf{S}_i$, and sending the decryption result to the adversary.

- **Challenge:** The adversary gives the challenger one attribute A^*, one group token $\mathcal{P}(\mathbf{G}^*)$ and two messages M_0, M_1, where $\mathcal{P}(\mathbf{G}^*)$ was generated in the query phase. The challenger responds by randomly choosing a coin $c \in \{0, 1\}$, generating a ciphertext $C^* \leftarrow \mathsf{ME}[A^*, \mathcal{P}(\mathbf{G}^*), M_c]$, and sending the challenge ciphertext to the adversary.
- **Phase 2:** The adversary can continue the query the same as Phase 1 except no decryption query on the challenge ciphertext C^* for $(A^*, \mathcal{P}(\mathbf{G}^*))$.
- **Win:** The adversary outputs a guess c' of c and wins the game if $c' = c$.

We define the advantage of adversary as $\mathsf{Adv}_{I_1} = \big|\Pr[c' = c] - 1/2\big|$.

Definition 2 *A membership encryption generated with a security parameter* 1^λ *is* (t, q_k, q_d, ϵ)*-secure against secret key if for all* t*-polynomial time adversaries who make* q_k *token key queries at most and* q_d *decryption queries at most, we have* $\epsilon = \mathsf{Adv}_{I_1}$ *is a negligible function of* λ*.*

Game 2: Indistinguishability against Membership.

- **Setup:** The challenger runs the Setup algorithm to generate the system parameter SP, and sends it to the adversary.
- **Challenge:** The adversary gives the challenger one attribute $A^*, \mathcal{P}(\mathbf{G}^*), \mathbf{G}^*$, $\mathbf{S}$ and two messages M_0, M_1. The challenger first verifies that $A \notin \mathcal{P}(\mathbf{G}^*)$ with $\mathbf{G}^*$ and $\mathbf{S}$. Then, the challenger responds by randomly choosing a coin $c \in \{0, 1\}$, generating a ciphertext $C^* \leftarrow \mathsf{ME}[A^*, \mathcal{P}(\mathbf{G}^*), M_c]$, and sending the challenge ciphertext to the adversary.
- **Win:** The adversary outputs a guess c' of c and wins the game if $c' = c$.

We define the advantage of adversary as $\mathsf{Adv}_{I_2} = \big|\Pr[c' = c] - 1/2\big|$.

Definition 3 *A membership encryption generated with a security parameter* 1^λ *is* (t, ϵ)*-secure against membership if for all* t*-polynomial time adversaries, we have* $\epsilon = \mathsf{Adv}_{I_2}$ *is a negligible function of* λ*. We call the membership encryption selectively secure [4] against membership if the adversary must output* A^* *and* $\mathbf{G}^*$ *before the setup of system parameters.*

Definition 4 (Privacy) *A membership encryption preserves the privacy of group attributes if given a group token* $\mathcal{P}(\mathbf{G})$ *and two group attributes* $\mathbf{G}_0 = \{A_1, A_2, \cdots, A_{k_1}\}$ *and* $\mathbf{G}_1 = \{A'_1, A'_2, \cdots, A'_{k_2}\}$*, it is computationally hard to decide whether* $\mathbf{G} = \mathbf{G}_0$ *or* $\mathbf{G} = \mathbf{G}_1$*.*

A secure membership encryption only guarantees the decryptor has the secret key $\mathbf{S}$ and A belongs to $\mathbf{G}$. To protect the privacy of group tokens, the membership encryption must capture the privacy property defined above. The game playing of privacy is defined as follows.

Game 3: Privacy.

- **Setup:** The challenger runs the Setup algorithm to generate the system parameter SP, and sends it to the adversary.

– **Challenge:** The adversary gives the challenger two group attributes $\mathbf{G}_0 = \{A_1, A_2, \cdots, A_{k_1}\}$ and $\mathbf{G}_1 = \{A'_1, A'_2, \cdots, A'_{k_2}\}$. The challenger responds by randomly choosing a coin $c \in \{0, 1\}$ and generating $\mathcal{P}(\mathbf{G}_c)$ for $\mathbf{G}_c$. Then, the challenger sends $\mathcal{P}(\mathbf{G}_c)$ to the adversary.
– **Win:** The adversary outputs a guess c' of c and wins the game if $c' = c$.

We define the advantage of adversary as $\mathsf{Adv}_P = \big|\Pr[c' = c] - 1/2\big|$.

Definition 5 *A membership encryption generated with a security parameter* 1^λ *preserves the privacy of group tokens with* (t, ϵ) *if for all* t*-polynomial time adversaries, we have* $\epsilon = \mathsf{Adv}_P$ *is a negligible function of* λ*. We say it unconditionally preserves the privacy of group tokens if* $\epsilon = 0$ *for any time* t *and* SP *is generated by the adversary.*

The properties of message security and privacy are sufficient for the definition of membership encryption. We define the additional maximum number accountability so as to apply it to constructing a flexible (K, N)-oblivious transfer protocol for any $K \leq N$.

Definition 6 (Maximum Number Accountability) *A membership encryption captures the property of maximum number accountability, if it is computationally hard to generate a group token pair* $(\mathcal{P}(\mathbf{G}), \mathbf{S})$ *for* $\mathbf{G}$ *with* k *attributes, but the verification shows that* $|\mathcal{P}(\mathbf{G})| < k$*.*

Game 4: Maximum Number Accountability.

– **Challenge:** The challenger runs the Setup algorithm to generate the system parameter SP, and sends it to the adversary.
– **Win:** The adversary outputs $(\mathcal{P}(\mathbf{G}^*), \mathbf{G}^*, \mathbf{S})$ and wins the game if $\mathbf{G}^*$ contains k numbers of attributes but the verification on $\mathcal{P}(\mathbf{G}^*)$ shows that it contains less than k attributes.

We define the advantage of adversary as Adv_A.

Definition 7 *A membership encryption generated with a security parameter* 1^λ *is* (t, ϵ)*-secure with maximum number accountability if for all* t*-polynomial time adversaries, we have* $\epsilon = \mathsf{Adv}_A$ *is a negligible function of* λ*.*

3 Our Membership Encryption

3.1 Pairing Group

Our membership encryption can be built from any pairing group. Let $\mathcal{G}_B$ be a generator of pairing groups. Taking as input a security parameter 1^λ, it outputs a pairing group $\mathbb{PG} = (\mathbb{G}, \mathbb{G}_T, e, p, g')$, where $\mathbb{G}, \mathbb{G}_T$ are two cyclic groups of prime order p, g' is a generator of $\mathbb{G}$, and $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is the bilinear map. The bilinear map e is a map with the following three properties:

– For all $u, v \in \mathbb{G}, a, b \in \mathbb{Z}_p$, $e(u^a, v^b) = e(u, v)^{ab}$.
– $e(g', g')$ is a generator of $\mathbb{G}_T$.
– It is efficient to compute the bilinear map $e(u, v)$ for any $u, v \in \mathbb{G}$.

3.2 The Scheme

Our group token generation is extended from the accumulator scheme in [17] with two secret keys α and β. Let $u \in \mathbb{G}$ and $u, u^{\alpha}, u^{\alpha^2}, \cdots, u^{\alpha^n}, u^{\beta\alpha}, u^{\beta\alpha^2}, \cdots, u^{\beta\alpha^n}$ be the components of system parameter. The group token $\mathcal{P}(\mathbf{G})$ for $\mathbf{G} = \{A_1, A_2, \cdots, A_k\} \in \mathbb{Z}_p$ is defined as

$$\mathcal{P}(\mathbf{G}) = (w_1, w_2, w_3) = \big(u^{\tau \prod_{i=1}^{k}(\alpha+A_i)},\ u^{\tau\beta \prod_{i=1}^{k}(\alpha+A_i)},\ u^{\tau\beta\alpha^{n-k} \prod_{i=1}^{k}(\alpha+A_i)}\big)$$

where $\mathbf{S} = \tau \in \mathbb{Z}_p$ is randomly chosen, and w_3 is the element for attribute number verification. Suppose $u^{\frac{1}{(\alpha+A)(\beta+A)}}$ is also in the system parameters and r is a randomness from $\mathbb{Z}_p$. Our approach for membership is described as follows:

- If $A \in \mathbf{G}$, we have $w_2 w_1^A$ contains $(\alpha + A)(\beta + A)$ such that

$$e\Big(\big(w_2 w_1^A\big)^r, u^{\frac{1}{(\alpha+A)(\beta+A)}}\Big) = e(u,u)^{r \prod_{A_i \in \mathbf{G}/A}(\alpha+A_i)}$$

  is computable from u^r and the system parameter.
- Otherwise, $A \notin \mathbf{G}$, we have $w_2 w_1^A = u^{\tau(\beta+A) \prod_{i=1}^{k}(\alpha+A_i)}$ such that

$$e\big((w_2 w_1^A)^r, u^{\frac{1}{(\alpha+A)(\beta+A)}}\big) = e(u,u)^{r \cdot \frac{\tau \prod_{A_i \in \mathbf{G}}(\alpha+A_i)}{\alpha+A}}$$

  contains the inversion exponent $\frac{1}{\alpha+A}$, which cannot be computed from u^r and the system parameter.

We use the above two different results to encrypt messages so that the decryption requires $A \in \mathbf{G}$. The detailed construction modified from [15] with security against chosen-plaintext attack ($q_d = 0$ in Game 1) is described as follows.

Setup: Taking as input a security parameter 1^λ, let n be the upper bound attribute number in group token generation and let all attributes be $\mathbb{A} = \{A_1, A_2, \cdots, A_n\} \subseteq \mathbb{Z}_p$, the setup algorithm works as follows:

- Choose a pairing group $\mathbb{PG} = (\mathbb{G}, \mathbb{G}_T, e, p, g')$.
- Choose $\alpha, \beta, \gamma \in \mathbb{Z}_p$ and $g, h \in \mathbb{G}$ at random. Compute $e(g^\gamma, h)$ and $g^{\gamma\alpha}$.
- Compute $u_i = h^{\gamma\alpha^i}$ and $v_i = h^{\gamma\beta\alpha^i}$ for all $i = 0, 1, \cdots, n$.
- Randomly choose s_i from $\mathbb{Z}_p$ and compute d_{A_i} for $i = 1, 2, \cdots, n$ as

$$d_{A_i} = \big(g^{\frac{s_i}{(\alpha+A_i)(\beta+A_i)}},\ \ h^{\frac{s_i-1}{\alpha}},\ \ h^{s_i},\ \ h^{s_i\alpha},\ \cdots,\ h^{s_i\alpha^{n-2}}\big).$$

The system parameter SP is defined as

$$SP = \Big(\mathbb{PG}, u_0, u_1, u_2, \cdots, u_n, v_0, v_1, v_2, \cdots, v_n, e(g^\gamma, h), g^{\gamma\alpha}, d_{A_1}, d_{A_2}, \cdots, d_{A_n}\Big).$$

GroupGen: Taking as input the group attribute $\mathbf{G} = \{A_1, A_2, \cdots, A_k\} \in \mathbb{Z}_p$ for any $k \leq n$, let $F(x) = \prod_{i=1}^{k}(x + A_i)$ and F_i be the coefficient of x^i, the

group token generation algorithm randomly chooses τ from $\mathbb{Z}_p$, sets $\mathbf{S} = \tau$ and computes $\mathcal{P}(\mathbf{G})$ as

$$\mathcal{P}(\mathbf{G}) = (w_1, w_2, w_3) = \Big(h^{\tau\gamma F(\alpha)}, \quad h^{\tau\gamma\beta F(\alpha)}, \quad h^{\tau\gamma\beta\alpha^{n-k}F(\alpha)}\Big)$$
$$= \Big(\prod_{i=0}^{k} u_i^{F_i\tau}, \quad \prod_{i=0}^{k} v_i^{F_i\tau}, \quad \prod_{i=0}^{k} v_{n-k+i}^{F_i\tau}\Big).$$

Verify: Taking as input $\mathcal{P}(\mathbf{G})$ and k, accept $|\mathcal{P}(\mathbf{G})| \leq k$ if $e(w_2, u_n) = e(w_3, u_k)$.

Encrypt: Taking as input an attribute $A \in \mathbb{A}$, a group token $\mathcal{P}(\mathbf{G}) = (w_1, w_2, w_3)$, a message $M \in \mathbb{G}_T$ and the system parameter, the encryption algorithm works as follows:

- Verify that $w_2 = w_1^{\beta}$ by checking $e(w_1, v_1) = e(w_2, v_0)$.
- Randomly choose r from $\mathbb{Z}_p$. Compute the ciphertext on the message M as

$$C = (c_1, c_2, c_3) = \Big(\big(w_2 w_1^A\big)^r, \quad (g^{\gamma\alpha})^r, \quad e(g^{\gamma}, h)^r \cdot M\Big).$$

Decrypt: Taking as input the ciphertext C, the secret key $\mathbf{S}$, the attribute A, the group attribute $\mathbf{G}$ and the system parameter, the decryption algorithm is described as follows.

- Compute

$$c_1' = c_1^{\frac{1}{\mathbf{S}}} = \Big(h^{r\tau\gamma(\beta+A)F(\alpha)}\Big)^{\frac{1}{\tau}} = h^{r\gamma(\beta+A)F(\alpha)}.$$

- Compute the pairing

$$e_1 = e\Big(c_1', \ g^{\frac{s}{(\alpha+A)(\beta+A)}}\Big) = e(g,h)^{rs\gamma\frac{F(\alpha)}{\alpha+A}}.$$

- If $A \in \mathbf{G}$, we have A is a root of $F(x)$. Let F_i' be the coefficient of x^i in $\frac{F(x)}{x+A}$. Compute the pairing

$$e_2 = e\Big((h^{\frac{s-1}{\alpha}})^{F_0'} \cdot \prod_{i=1}^{k-1} (h^{s\alpha^{i-1}})^{F_i'}, c_2\Big) = e(g,h)^{rs\gamma\frac{F(\alpha)}{\alpha+A} - F_0' r\gamma}.$$

- Compute M by

$$c_3 \cdot \Big(e_2 e_1^{-1}\Big)^{\frac{1}{F_0'}} = e(g^{\gamma}, h)^r M \cdot \Big(e(g,h)^{-F_0' r\gamma}\Big)^{\frac{1}{F_0'}} = M.$$

3.3 Discussions

The above membership encryption is proposed for security against chosen-plaintext attack (CPA), i.e., the adversary cannot make decryption queries. Let $\mathcal{ME}[M, r]$ be our ciphertext on M encrypted with the randomness r. Using the Fujisaki-Okamoto approach [13] in the random oracle model, we can easily extend it to

the security against chosen-ciphertext attack (CCA). Let $H_1 : \{0,1\}^* \to \mathbb{Z}_p$ and $H_2 : \{0,1\}^* \to \{0,1\}^{l_m}$ be cryptographic hash functions, where l_m denotes the length of messages. If $\mathcal{ME}[M, r]$ is secure against CPA, the following membership encryption construction for $(A, \mathcal{P}(\mathbf{G}))$ is secure against CCA

$$\mathcal{ME}\big[\sigma,\ H_1(A, \mathcal{P}(\mathbf{G}), \sigma, M)\big],\quad H_2(\sigma) \oplus M.$$

It is not hard to prove CCA security under our security model definition in Game 1 with the proof in [13]. We omit it here.

Our membership encryption captures the following nice features.

- Constant-size $\mathcal{P}(\mathbf{G})$. Our group token $\mathcal{P}(\mathbf{G})$ consists of three group elements only independent of the number of attributes in $\mathbf{G}$.
- Maximum number accountability. According to our setting, we have

  $$\mathcal{P}(\mathbf{G}) = (w_1, w_2, w_3) = (w_1, w_1^{\beta}, w_1^{\beta\alpha^{n-k}})$$

  for k numbers of attributes. Through the verification, the verifier knows that the exponent of w_3 contains α^{n-k}. We have $F(\alpha)$ in $w_1 = h^{\tau\gamma F(\alpha)}$ has k degrees at most; otherwise, computing w_3 needs $h^{\gamma\beta}, h^{\gamma\beta\alpha^1}, \cdots, h^{\gamma\beta\alpha^{n'}}$ for $n' > n$ and they are not given in the system parameter.
- Constant-size ciphertext. Our ciphertext is constant-size and is composed of two group elements from $\mathbb{G}$ and one group element from $\mathbb{G}_T$. The length of ciphertext depends on the length of security parameter only.

4 Proof of Security

In this section, we prove the security of our membership encryption. Before the security analysis, we introduce three hard problems adopted in our reduction proof.

4.1 Hard Problems

Our membership encryption uses a pairing group as an ingredient and its security relies on the hardness of three problems that are slightly modified from the GDDHE problem [12], the a-MSE-DDH problem [15] and the DHE problem [7]. The new hard problems are (f, n)-GDDHE problem that is adopted to prove message security against secret key (Game 1), (f, g, n)-aMSE-DDHE problem which is used to prove message security against membership (Game 2), and (f, n)-DHE problem that is used to prove the property of maximum number accountability (Game 4). We notice that the intractability of these three hard problems can be analysed in the generic group model by following the proof in [5, 12] for the original GDDHE problem. For completeness, we analyse these problems based on the Theorem 2 in [12] in the full version of this paper.

Let g_0, h_0, w be random generators from $\mathbb{G}$ and a, γ be random integers from $\mathbb{Z}_p$. The three hard problems are defined as follows.

(f, n)-GDDHE Problem:

Instance:	Any $(2n+1)$-degree polynomial function $f(x) \in \mathbb{Z}_p[x]$.
	$g_0,\ g_0^a,\ g_0^{a^2},\ \cdots,\ g_0^{a^{2n}},\quad g_0^{af(a)}\quad g_0^{af(a)\theta}$.
	$h_0,\ h_0^a,\ h_0^{a^2},\ \cdots,\ h_0^{a^{2n}}$
	$w,\ w^a,\ w^{a^2},\ \cdots,\ w^{a^{2n}},\quad w^\theta\quad w^{a\theta}$
	$T \in \mathbb{G}_T$, which is either at random or equal to $e(g_0, h_0)^{f(a)\theta}$
Target:	Return $b = 1$ if $T = e(g_0, h_0)^{f(a)\theta}$; otherwise, $b = 0$.

Definition 8 *The (f, n)-GDDHE problem holds with (t, ϵ) if given an instance generated from a security parameter 1^λ and any $(2n+1)$-degree polynomial function $f(x) \in \mathbb{Z}_p[x]$, the advantage of solving this problem in t polynomial time is ϵ at most which is a negligible function of λ.*

(f, g, n)-aMSE-DDH Problem:

Instance:	Any $(2n+1)$-degree polynomial function $f(x) \in \mathbb{Z}_p[x]$, and any degree $\leq 2n$ polynomial function $g(x) \in \mathbb{Z}_p[x]$, such that $gcd(f(x), g(x)) = 1$ (or any nonzero number).
	$g_0,\quad g_0^a,\quad g_0^{a^2},\ \cdots,\quad g_0^{a^{2n}},\quad g_0^{\theta af(a)}$
	$g_0^\gamma,\quad g_0^{\gamma a},\quad g_0^{\gamma a^2},\ \cdots,\quad g_0^{\gamma a^{2n+2}}$
	$h_0,\quad h_0^a,\quad h_0^{a^2},\cdots,\quad h_0^{a^{2n}},\quad h_0^{\theta g(a)}$
	$h_0^\gamma,\quad h_0^{\gamma a},\quad h_0^{\gamma a^2},\cdots,\quad h_0^{\gamma a^{2n}}$
	$T \in \mathbb{G}_T$, which is either at random or equal to $e(g_0, h_0)^{f(a)\theta}$
Target:	Return $b = 1$ if $T = e(g_0, h_0)^{f(a)\theta}$; otherwise, $b = 0$.

Definition 9 *The (f, g, n)-aMSE-DDH problem holds with (t, ϵ) if given an instance generated from a security parameter 1^λ and any two co-prime polynomial functions $f(x), g(x)$ in $\mathbb{Z}_p[x]$ with $(2n+1)$ degrees and $\leq 2n$ degrees respectively, the advantage of solving this problem in t polynomial time is ϵ at most which is a negligible function of λ.*

(f, n)-DHE Problem:

Instance:	$g_0,\ g_0^a,\ g_0^{a^2},\ \cdots,\ g_0^{a^n}$.
Output:	Return $(f(x), g_0^{f(a)})$, where $f(x) \in \mathbb{Z}_p[x]$ is an n'-degree polynomial function $n' > n$.

Definition 10 *The (f, n)-DHE problem holds with (t, ϵ) if given an instance generated from a security parameter 1^λ, the advantage of solving this problem in t polynomial time is ϵ at most which is a negligible function of λ.*

4.2 Security Proof

Theorem 1 (Indistinguishability against Secret Key). *Suppose the (f, n)-GDDHE problem is (t, ϵ)-hard, we can construct (t', q_k, ϵ')-secure membership encryption against secret key. Here, $t' = t - O(q_k n t_e)$ and $\epsilon' = \frac{\epsilon}{q_k}$, where t_e denotes the average time of an exponentiation in $\mathbb{G}$.*

The proof is given in the full version of this paper.

Theorem 2 (Indistinguishability against Membership). *Suppose the* (f, g, n)*-aMSE-DDH problem is* (t, ϵ)*-hard, we can construct* (t', q_k, ϵ') *selectively secure membership encryption against membership. Here,* $t' = t - O(n^2 t_e)$ *and* $\epsilon' = \epsilon$*, where* t_e *denotes the average time of an exponentiation in* $\mathbb{G}$*.*

We prove the security of membership in the selective security model in which the adversary must output A^* and $\mathbf{G}^*$ before the setup of system parameter. The security proof can be transformed into full security by correctly guessing the challenge target, but it is only suitable for small n and $\mathbf{G}^*$.

Proof. Suppose there exists an adversary who can break the membership encryption against membership under selective security model. We construct an algorithm $\mathcal{B}$ that solves the (f, g, n)-aMSE-DDH problem. $\mathcal{B}$ interacts with the adversary as the follows.

Initialization. Let $\mathbb{PG} = (\mathbb{G}, \mathbb{G}_T, e, p, g')$ be the pairing group and $\mathbb{A} = \{A_1, A_2, \cdots, A_n\}$ be all attributes. The adversary outputs $(A^*, \mathbf{G}^*)$ for challenge where $A^* \notin \mathbf{G}^*$.

Setup. The algorithm $\mathcal{B}$ works as follows to simulate the system parameter.

- Let $\mathbf{G}^* = \{A_1^*, A_2^*, \cdots, A_k^*\}$ be the attributes in $\mathbf{G}^*$. Define the set $\mathbf{G}_1$ as follows
$$\mathbf{G}_1 = \{A_1, A_2, \cdots, A_n\}/\{A_1^*, A_2^*, \cdots, A_k^*, A^*\}.$$
- Randomly choose β_0, β_1 from $\mathbb{Z}_p$. Let $f(x)$ be a $(2n+1)$-degree polynomial function and $g(x)$ be a $(k+1)$-degree polynomial function defined as
$$(x + A^*) \prod_{A_i \in \mathbf{G}_1} (x + A_i) \cdot \prod_{A_i \in \mathbb{A}} (\beta_0 x + \beta_1 + A_i) \Big| f(x)$$
$$g(x) = (\beta_0 x + \beta_1 + A^*) \prod_{A_i \in \mathbf{G}^*} (x + A_i),$$
such that $gcd\big(f(x), g(x)\big) = 1$ (or any nonzero number).
- Send $f(x), g(x)$ to the (f, g, n)-aMSE-DDH problem generator. Let be challenge instance be
$$\begin{array}{llllll} g_0, & g_0^a, & g_0^{a^2}, & \cdots, & g_0^{a^{2n}}, & g_0^{\theta a f(a)} \\ g_0^\gamma, & g_0^{\gamma a}, & g_0^{\gamma a^2}, & \cdots, & g_0^{\gamma a^{2n+2}} & \\ h_0, & h_0^a, & h_0^{a^2}, & \cdots, & h_0^{a^{2n}}, & h_0^{\theta g(a)} \\ h_0^\gamma, & h_0^{\gamma a}, & h_0^{\gamma a^2}, & \cdots, & h_0^{\gamma a^{2n}} & \\ T \in \mathbb{G}_T & & & & & \end{array}$$
- Set $\alpha, \beta, \gamma, g, h$ as
$$\alpha = a, \quad \beta = \beta_0 a + \beta_1, \quad \gamma = \gamma, \quad g = g_0^{f(a)}, \quad h = h_0,$$
where a, γ are the randomness in the challenge instance.

– Compute $e(g^\gamma, h), g^{\gamma\alpha}, u_i, v_i$, as

$$e(g^\gamma, h) = e(g_0, h_0)^{\gamma f(a)}, \quad g^{\gamma\alpha} = g_0^{af(a)\gamma}, \quad u_i = h^{\gamma\alpha^i} = h_0^{\gamma a^i}$$

$$v_i = h^{\gamma\beta\alpha^i} = h_0^{\beta_0\gamma a^{i+1} + \beta_1\gamma a^i}.$$

– Compute d_{A_i} as follows.

- Randomly choose s_i' from $\mathbb{Z}_p$ and set

$$s_i = (s_i'\gamma a + 1) f_{A_i}(a),$$

where $f_{A_i}(x)$ is defined as follows

$$f_{A_i}(x) = \begin{cases} \frac{1}{\beta_1 + A^*}(\beta_0 x + \beta_1 + A^*) & \text{if } A_i = A^*, \\ \frac{1}{A_i}(x + A_i) & \text{else if } A_i \in \mathbf{G}^*, \\ 1 & \text{otherwise } A_i \in \mathbf{G}_1. \end{cases}$$

We have

$$\frac{f_{A_i}(a) - 1}{a} = \begin{cases} \frac{\beta_0}{\beta_1 + A^*} & \text{if } A_i = A^*, \\ \frac{1}{A_i} & \text{else if } A_i \in \mathbf{G}^*, \\ 0 & \text{otherwise } A_i \in \mathbf{G}_1. \end{cases}$$

such that

$$\begin{aligned} \frac{s_i - 1}{\alpha} &= \frac{(s_i'\gamma a + 1) f_{A_i}(a) - 1}{a} = f'_{A_i}(a) \\ &= \gamma s_i' f_{A_i}(a) + \frac{f_{A_i}(a) - 1}{a} = a_2\gamma a + a_1\gamma + a_0, \end{aligned}$$

where a_2, a_1, a_0 are coefficients. Let $f''_{A_i}(x)$ be defined as

$$f''_{A_i}(x) = \frac{f(x) f_{A_i}(x)}{(x + A_i)(\beta_0 x + \beta_1 + A_i)}.$$

We have $f''_{A_i}(x)$ is a polynomial function with $2n$ degrees at most.

- Compute d_{A_i} as

$$\left(g_0^{s_i'\gamma a f''_{A_i}(a) + f''_{A_i}(a)}, \ h_0^{f'_{A_i}(a)}, \ h_0^{(s_i'\gamma a + 1) f_{A_i}(a)}, \cdots, h_0^{(s_i'\gamma a + 1) f_{A_i}(a) a^{n-2}}\right).$$

According to the setting of the randomness $s_i = (s_i'\gamma a + 1) f_{A_i}(a)$, we have

$$\begin{aligned} d_{A_i} &= \left(g_0^{s_i'\gamma a f''_{A_i}(a) + f''_{A_i}(a)}, \ h_0^{f'_{A_i}(a)}, \ h_0^{(s_i'\gamma a + 1) f_{A_i}(a)}, \cdots, h_0^{(s_i'\gamma a + 1) f_{A_i}(a) a^{n-2}}\right) \\ &= \left(g^{\frac{s_i}{(\alpha + A_i)(\beta + A_i)}}, \ h^{\frac{s_i - 1}{\alpha}}, \ h^{s_i}, \ \cdots, \ h^{s_i\alpha^{n-2}}\right). \end{aligned}$$

All elements are computable from the challenge instance and setting. $\mathcal{B}$ generates the system parameter and sends it to the adversary.

Challenge. The adversary returns $(A^*, \mathcal{P}(\mathbf{G})^*, \mathbf{G}^*, \mathbf{S}^*, M_0, M_1)$ for challenge. Let $\mathbf{S} = \tau^*$, $\mathcal{P}(\mathbf{G}^*) = (w_1, w_2, w_3)$ and $\mathbf{G}^* = \{A_1^*, A_2^*, \cdots, A_k^*\}$. The algorithm $\mathcal{B}$ randomly chooses a coin $c \in \{0, 1\}$, and simulates the challenge ciphertext as follows

$$C = (c_1^*, c_2^*, c_3^*) = \Big(\big(h_0^{\theta g(a)}\big)^{\tau^*}, \quad g_0^{\theta a f(a)}, \quad T \cdot M_c \Big).$$

Let $r = \frac{\theta}{\gamma}$. If $T = e(g_0, h_0)^{\theta f(a)}$, we have

$$\begin{aligned}
(w_2 w_1^{A^*})^r &= (h^{\tau^* \gamma \prod_{i=1}^{k} (\alpha + A_i^*)(\beta + A^*)})^r = (h_0^{\tau^* \gamma g(a)})^{\frac{\theta}{\gamma}} = \big(h_0^{\theta g(a)}\big)^{\tau^*} \\
(g^{\gamma\alpha})^r &= (g_0^{\gamma a f(a)})^{\frac{\theta}{\gamma}} = g_0^{\theta a f(a)} \\
e(g^\gamma, h)^r &= e(g_0^{\gamma f(a)}, h_0)^{\frac{\theta}{\gamma}} = e(g_0, h_0)^{\theta f(a)} = T.
\end{aligned}$$

Therefore, $C = (c_1^*, c_2^*, c_3^*)$ is a valid ciphertext on M_c for $(A^*, \mathcal{P}(\mathbf{G}^*))$. $\mathcal{B}$ sends it to the adversary.

Win: The adversary outputs $c' \in \{0, 1\}$, and the algorithm $\mathcal{B}$ outputs c' as the guess of T.

This completes the description of our simulation. If $T = e(g_0, h_0)^{\theta f(a)}$, the challenge ciphertext is valid and the adversary will output $c' = c$ with advantage $1/2 + \epsilon$; otherwise, T is universally random and the adversary's advantage is $1/2$. The simulation time is mainly dominated by the $d_{\mathbb{A}}$ simulation, and each d_{A_i} costs $O(n)$ exponentiations. No abortion occurs during our simulation. We therefore obtain the Theorem 2. □

Theorem 3 (Privacy). *$\mathcal{P}(\mathbf{G})$ unconditionally preserves the privacy of all attributes in $\mathbf{G}$.*

Proof. Let $\mathcal{P}(\mathbf{G})$ be a group token generated from $\mathbf{G} = \{A_1, A_2, \cdots, A_{k_1}\}$ and $\mathbf{S} = \tau$. We have

$$\mathcal{P}(\mathbf{G}) = (w_1, w_2, w_3) = \Big(h^{\tau\gamma \prod_{i=1}^{k_1} (\alpha + A_i)}, \quad h^{\tau\gamma\beta \prod_{i=1}^{k_1} (\alpha + A_i)}, \quad h^{\tau\beta\alpha^{n-k_1} \prod_{i=1}^{k_1} (\alpha + A_i)} \Big).$$

Since there exists $\mathbf{G}' = \{A_1', A_2', \cdots, A_{k_2}'\}$ and $\tau' \in \mathbb{Z}_p$ satisfying

$$\tau \prod_{i=1}^{k_1} (\alpha + A_i) = \tau' \prod_{i=1}^{k_2} (\alpha + A_i'),$$

we have $\mathcal{P}(\mathbf{G})$ can be also seen as a group token generated for $\mathbf{G}' = \{A_1', A_2', \cdots, A_{k_2}'\}$ and τ'. Thus, the privacy of all attributes in $\mathcal{P}(\mathbf{G})$ is unconditionally preserved. This completes the proof and we obtain the Theorem 3. □

Theorem 4 (Maximum Number Accountability). *Suppose the (f, n)-DHE problem is hard, the group token $\mathcal{P}(\mathbf{G})$ is secure with maximum number accountability.*

The proof is given in the full version of this paper.

5 Oblivious Transfer from Membership Encryption

In this section, we show how to construct an efficient K-out-of-N oblivious transfer protocol ($\mathsf{OT}^{\mathsf{K}}_{\mathsf{N}}$) from membership encryption. Our $\mathsf{OT}^{\mathsf{K}}_{\mathsf{N}}$ protocol only requires two rounds between receiver and sender. Using our construction, the $\mathsf{OT}^{\mathsf{K}}_{\mathsf{N}}$ protocol exhibits the nice property of constant communication cost, where the receiver sends constant-size messages to the sender independent of K and N.

Suppose the sender has messages $M_1, M_2, M_3, \cdots, M_N$ for any $N \leq n$, and the receiver wants to receive messages $M_{i_1}, M_{i_2}, \cdots, M_{i_K}$ for any $\{i_1, i_2, \cdots, i_K\} \subseteq \{1, 2, \cdots, N\}$. Let SP be the system parameter of membership encryption, where all attributes are the indices, i.e. $\mathbb{A} = \{1, 2, \cdots, n\}$. Our OT protocol from membership encryption depicted in **Fig. 1** is described as follows.

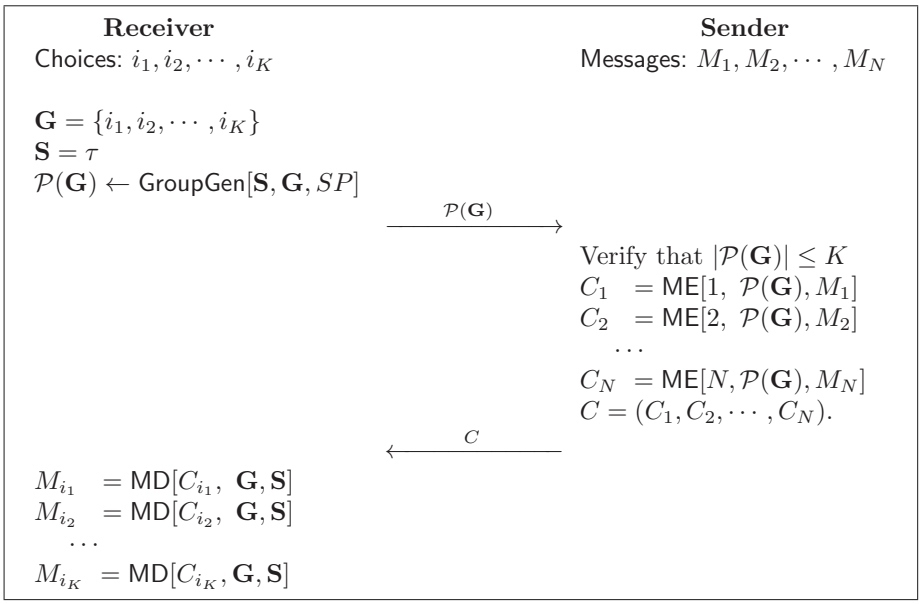

Fig. 1. K-Out-of-N Oblivious Transfer.

- The receiver runs the $\mathsf{GroupGen}$ algorithm to generate $\mathcal{P}(\mathbf{G})$ on $\mathbf{G} = \{i_1, i_2, \cdots, i_K\}$, and sends $\mathcal{P}(\mathbf{G})$ to the sender.
- Upon receiving $\mathcal{P}(\mathbf{G})$ from the receiver, the sender verifies that $|\mathcal{P}(\mathbf{G})| \leq K$. If it is false, reject. Otherwise, runs the encryption algorithm $C_i = \mathsf{ME}[i, \mathcal{P}(\mathbf{G}), M_i]$ for all $i = 1, 2, \cdots, N$ and sends all ciphertexts $C = (C_1, C_2, \cdots, C_N)$ to the receiver.
- Upon receiving all ciphertexts from the sender, the receiver runs the decryption algorithm $\mathsf{MD}[C_i, \mathbf{G}, \mathbf{S}]$ to get the message M_i for all $i = i_1, i_2, \cdots, i_K$.

Our $\mathsf{OT}^{\mathsf{K}}_{\mathsf{N}}$ scheme preserves receiver's privacy and protects sender's messages against malicious receivers under the security model definition in [10]. According

to the Theorem 3, given $\mathcal{P}(\mathbf{G})$, the sender cannot distinguish $\mathcal{P}(\mathbf{G})$ which is generated from either $\mathbf{G} = \{i_1, i_2, \cdots, i_K\}$ or $\mathbf{G} = \{i'_1, i'_2, \cdots, i'_K\}$. According to the Theorems 2 and 4, the receiver can only obtain chosen messages M_j for all $j \in \{i_1, i_2, \cdots, i_K\}$.

The system parameter SP in our OT protocol can be generated by the sender or the trust third party for the universal application. In our $\mathsf{OT}^{\mathsf{K}}_{\mathsf{N}}$ scheme, the receiver sends the group token $\mathcal{P}(\mathbf{G})$ aggregated for all K choices and the sender responds with N ciphertexts. The $\mathsf{OT}^{\mathsf{K}}_{\mathsf{N}}$ protocol is composed of two rounds only. Using our membership encryption, our group token is constant-size and is independent of the number of choice K, and our ciphertext is also constant-size and is dependent on the security parameter only. In **Table 1**, we compare the communication cost of all two-round K-out-of-N oblivious transfer protocols in the literature. It shows that $\mathsf{OT}^{\mathsf{K}}_{\mathsf{N}}$ protocol from our membership encryption requires the smallest communication cost.

Table 1. Communication cost of two-round K-out-of-N oblivious transfer.

	[16]	[18, 10, 9]	**Ours**
Messages from Receiver to Sender	$O(N)$	$O(K)$	$O(1)$
Messages from Sender to Receiver	$O(N)$	$O(N)$	$O(N)$

6 Conclusion

Protecting membership privacy is essential in many applications. Existing solutions were based on membership proof for $\mathcal{P}(A) \in \mathbf{G}$ and $A \in \mathcal{P}(\mathbf{G})$. In this work, we extended the membership proof to membership encryption. We introduced the notion of membership encryption, where if the encryption takes as input an attribute A and a privacy-preserving group token $\mathcal{P}(\mathbf{G})$, successful decryption must satisfy $A \in \mathbf{G}$. We constructed a provably secure membership encryption where the group token $\mathcal{P}(\mathbf{G})$ is constant-size and the maximum attribute number is accountable. The ciphertext is also constant and is dependent on security parameter only. We showed how to apply our encryption scheme to the construction of two-round K-out-of-N oblivious transfer protocols $\mathsf{OT}^{\mathsf{K}}_{\mathsf{N}}$. Using our membership encryption, the $\mathsf{OT}^{\mathsf{K}}_{\mathsf{N}}$ protocol only requires $O(1)$ communication cost from receiver to sender, against the other existing two-round oblivious transfer protocols.

Acknowledgement. We would like to thank the anonymous reviewers for their helpful comments and suggestions.

References

1. Au, M.H., Tsang, P.P., Susilo, W., Mu, Y.: Dynamic universal accumulators for DDH groups and their application to attribute-based anonymous credential systems. In: Fischlin, M. (ed.) CT-RSA 2009. LNCS, vol. 5473, pp. 295–308. Springer, Heidelberg (2009)

2. Bari, N., Pfitzmann, B.: Collision-free accumulators and fail-stop signature schemes without trees. In: Fumy, W. (ed.) EUROCRYPT 1997. LNCS, vol. 1233, pp. 480–494. Springer, Heidelberg (1997)
3. Benaloh, J.C., de Mare, M.: One-way accumulators: A decentralized alternative to digital sinatures (extended abstract). In: Helleseth, T. (ed.) EUROCRYPT 1993. LNCS, vol. 765, pp. 274–285. Springer, Heidelberg (1993)
4. Boneh, D., Boyen, X.: Efficient selective-id secure identity-based encryption without random oracles. In: Cachin, C., Camenisch, J. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 223–238. Springer, Heidelberg (2004)
5. Boneh, D., Boyen, X., Goh, E.J.: Hierarchical identity based encryption with constant size ciphertext. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 440–456. Springer, Heidelberg (2005)
6. Camenisch, J., Chaabouni, R., Shelat, A.: Efficient protocols for set membership and range proofs. In: Pieprzyk, J. (ed.) ASIACRYPT 2008. LNCS, vol. 5350, pp. 234–252. Springer, Heidelberg (2008)
7. Camenisch, J., Kohlweiss, M., Soriente, C.: An accumulator based on bilinear maps and efficient revocation for anonymous credentials. In: Jarecki, S., Tsudik, G. (eds.) Public Key Cryptography 2009. LNCS, vol. 5443, pp. 481–500. Springer, Heidelberg (2009)
8. Camenisch, J., Lysyanskaya, A.: Dynamic accumulators and application to efficient revocation of anonymous credentials. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 61–76. Springer, Heidelberg (2002)
9. Chen, Y., Chou, J.S., Hou, X.W.: A novel k-out-of-n oblivious transfer protocols based on bilinear pairings. IACR Cryptology ePrint Archive 2010, 27 (2010)
10. Chu, C.K., Tzeng, W.G.: Efficient k-out-of-n oblivious transfer schemes with adaptive and non-adaptive queries. In: Vaudenay, S. (ed.) Public Key Cryptography 2005. LNCS, vol. 3386, pp. 172–183. Springer, Heidelberg (2005)
11. Cramer, R., Damgård, I., Schoenmakers, B.: Proofs of partial knowledge and simplified design of witness hiding protocols. In: Desmedt, Y. (ed.) CRYPTO 1994. LNCS, vol. 839, pp. 174–187. Springer, Heidelberg (1994)
12. Delerablée, C.: Identity-based broadcast encryption with constant size ciphertexts and private keys. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 200–215. Springer, Heidelberg (2007)
13. Fujisaki, E., Okamoto, T.: Secure integration of asymmetric and symmetric encryption schemes. In: Wiener, M.J. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 537–554. Springer, Heidelberg (1999)
14. Guo, F., Mu, Y., Chen, Z.: Mutative identity-based signatures or dynamic credentials without random oracles. In: Bao, F., Ling, S., Okamoto, T., Wang, H., Xing, C. (eds.) CANS 2007. LNCS, vol. 4856, pp. 1–14. Springer, Heidelberg (2007)
15. Herranz, J., Laguillaumie, F., Ràfols, C.: Constant size ciphertexts in threshold attribute-based encryption. In: Nguyen, P.Q., Pointcheval, D. (eds.) Public Key Cryptography 2010. LNCS, vol. 6056, pp. 19–34. Springer, Heidelberg (2010)
16. Mu, Y., Zhang, J., Varadharajan, V.: m out of n oblivious transfer. In: Batten, L.M., Seberry, J. (eds.) ACISP 2002. LNCS, vol. 2384, pp. 395–405. Springer, Heidelberg (2002)
17. Nguyen, L.: Accumulators from bilinear pairings and applications. In: Menezes, A. (ed.) CT-RSA. LNCS, vol. 3376, pp. 275–292. Springer, Heidelberg (2005)
18. Ogata, W., Kurosawa, K.: Oblivious keyword search. J. Complexity 20(2-3), 356–371 (2004)
19. Rabin, M.O.: How to exchange secrets with oblivious transfer. IACR Cryptology ePrint Archive 2005, 187 (2005)

Appendix A: Analysis of Hard Problems

Our intractability analysis is based on the proof given in [12] (Theorem 2). We give the sketch here only, and refer readers to [12] for detailed proof by combining the following analysis.

In the definition of (f, n)-DHE problem, no coefficients $y_0, y_1, y_2, \cdots, y_{n'}$ exist such that $f(a) = y_0 + y_1 a + y_2 a^2 + \cdots + y_{n'} a^{n'}$ is a polynomial function in $\mathbb{Z}_p[a]$ with $n' > n$. The (f, n)-DHE problem therefore is hard in the generic group model.

In the definition of (f, g, n)-aMSE-DDHE problem, let $h_0 = g_0^{\phi}$, the polynomial function $f(a)\theta\phi$ cannot be combined from

$$\begin{array}{llllll} \gamma, & \gamma a, & \gamma a^2, & \cdots, & \gamma a^{2n+2}, \\ \gamma\phi, & \gamma a\phi, & \gamma a^2\phi, & \cdots, & \gamma a^{2n}\phi \end{array}$$

due to the unknown γ. The remained possible combination is from

$$\begin{array}{llllll} & a, & a^2, & \cdots, & a^{2n}, & af(a)\theta \\ \phi, & a\phi, & a^2\phi, & \cdots, & a^{2n}\phi & g(a)\theta\phi \end{array}$$

denoted by

$$f(a)\theta\phi = A(a) \cdot g(a)\theta\phi + B(a)\phi \cdot af(a)\theta,$$

where $A(a), B(a)$ are any $2n$-degree polynomial functions in $\mathbb{Z}_p[a]$. This case is similar to the analysis in Theorem 2 [12], which shows that $f(a)\theta\phi$ is independent of the given instance. Therefore, the (f, g, n)-aMSE-DDH problem is hard in the generic group model.

In the definition of (f, n)-GDDHE problem, let $h_0 = g_0^{\phi}$ and $w = g_0^{\psi}$, the polynomial function $f(a)\theta\phi$ cannot be combined from $\psi, \psi a, \psi a^2, \cdots, \psi a^n, \psi\theta, \psi a\theta$ due to the unknown ψ. The remained possible combination is the same as the original GDDHE problem defined in [12], and the analysis shows that (f, n)-GDDHE problem is hard.

Appendix B: Proof of Theorem 1

Proof. Suppose there exists an adversary who can break the membership encryption against secret key by a chosen-plaintext attack. We construct an algorithm $\mathcal{B}$ that solves the (f, n)-GDDHE problem. $\mathcal{B}$ interacts with the adversary as the follows.

Setup. Let $\mathbb{PG} = (\mathbb{G}, \mathbb{G}_T, e, p, g')$ be the pairing group and $\mathbb{A} = \{A_1, A_2, \cdots, A_n\}$ be all attributes. The algorithm $\mathcal{B}$ works as follows to simulate the system parameter.

– Pick random $\beta_0, \beta_1, \gamma_0 \in \mathbb{Z}_p$ and set $\alpha, \beta, \gamma, g, h$ as

$$\alpha = a, \quad \beta = \beta_0 a + \beta_1, \quad \gamma = \gamma_0, \quad g = g_0^{f(a)}, \quad h = h_0,$$

where $\mathcal{B}$ queries an instance of (f, n)-GDDHE problem satisfying

$$\prod_{i=1}^{n}(x + A_i)(\beta_0 x + \beta_1 + A_i) \Big| f(x).$$

– Compute $e(g^\gamma, h), g^{\gamma\alpha}, u_i, v_i$, as

$$e(g^\gamma, h) = e(g_0, h_0)^{\gamma_0 f(a)}, \quad g^{\gamma\alpha} = g_0^{\gamma_0 a f(a)}, \quad u_i = h^{\gamma\alpha^i} = h_0^{\gamma_0 a^i}$$

$$v_i = h^{\gamma\beta\alpha^i} = h_0^{\gamma_0\beta_0 a^{i+1} + \gamma_0\beta_1 a^i}.$$

– For $i = 1, 2, \cdots, n$, randomly choose $s_i' \in \mathbb{Z}_p$, and compute d_{A_i} as

$$d_{A_i} = \left(g_0^{(s_i'a+1)f_{A_i}(a)},\ h_0^{s_i'},\ h_0^{(s_i'a+1)a}, \cdots, h_0^{(s_i'a+1)a^{n-2}}\right).$$

Let $s_i = s_i'a + 1$ and $f_{A_i}(x)$ be

$$f_{A_i}(x) = \frac{f(x)}{(x + A_i)(\beta_0 x + \beta_1 + A_i)},$$

which is a $(2n-1)$-degree polynomial function. We have

$$\begin{aligned} d_{A_i} &= \left(g_0^{(s_i'a+1)f_{A_i}(a)},\ h_0^{s_i'},\ h_0^{s_i'a+1},\ h_0^{(s_i'a+1)a}, \cdots, h_0^{(s_i'a+1)a^{n-2}}\right) \\ &= \left(g^{\frac{s_i}{(\alpha+A_i)(\beta+A_i)}},\ h^{\frac{s_i-1}{\alpha}},\ h^{s_i}, \cdots,\ h^{s_i\alpha^{n-2}}\right). \end{aligned}$$

All elements are computable from the challenge instance and setting. $\mathcal{B}$ generates the system parameter and sends it to the adversary.

Phase 1. The algorithm $\mathcal{B}$ randomly chooses i^* from $[1, n]$ and simulate the group token as follows:

– For a group token query on $\mathbf{G}_i$, if $i \neq i^*$, run the $\mathsf{GroupGen}$ algorithm to generate $\mathcal{P}(\mathbf{G}_i)$.
– Otherwise $i = i^*$, let $|\mathbf{G}_{i^*}| = k$, compute $\mathcal{P}(\mathbf{G}_{i^*})$ as

$$\begin{aligned} \mathcal{P}(\mathbf{G}^*) &= (w_1, w_2, w_3) = (w,\ w^\beta,\ w^{\beta\alpha^{n-k}}) \\ &= \left(w,\ w^{a\beta_0} \cdot w^{\beta_1},\ \ w^{\beta_0 a^{n-k+1}} \cdot w^{\beta_1 a^{n-k}}\right), \end{aligned}$$

which is computable from the challenge instance.

Challenge. The adversary returns $(A^*, \mathcal{P}(\mathbf{G}^*), M_0, M_1)$ for challenge. If $\mathbf{G}^* \neq \mathbf{G}_{i^*}$, abort; otherwise, $\mathcal{P}(\mathbf{G}^*) = (w_1, w_2, w_3) = (w, w^\beta, w_3)$. The algorithm $\mathcal{B}$ randomly chooses a coin $c \in \{0, 1\}$, and simulates the challenge ciphertext as follows

$$C = (c_1^*, c_2^*, c_3^*) = \Big((w^{a\theta})^{\beta_0} \cdot (w^\theta)^{\beta_1 + A^*},\ \ (g_0^{af(a)\theta})^{\gamma_0},\ \ T^{\gamma_0} \cdot M_c\Big).$$

Let $r = \theta$, if $T = e(g_0, h_0)^{f(a)\theta}$, we have

$$\begin{aligned}(w_2 w_1^{A^*})^r &= (w^{\beta_0 a + \beta_1 + A^*})^\theta = (w^{a\theta})^{\beta_0} \cdot (w^\theta)^{\beta_1 + A^*} \\ (g^{\alpha\gamma})^r &= (g_0^{af(a)\gamma_0})^\theta = (g_0^{af(a)\theta})^{\gamma_0} \\ e(g^\gamma, h)^r \cdot M_c &= e(g_0^{\gamma_0 f(a)}, h_0)^\theta = T^{\gamma_0} \cdot M_c.\end{aligned}$$

Therefore, $C = (c_1^*, c_2^*, c_3^*)$ is a valid ciphertext on M_c for $(A^*, \mathcal{P}(\mathbf{G}^*))$. $\mathcal{B}$ sends it to the adversary.

Win: The adversary outputs $c' \in \{0, 1\}$. Then, the algorithm $\mathcal{B}$ outputs c' as the guess of T.

This completes the description of our simulation. If $T = e(g_0, h_0)^{\theta f(a)}$, the challenge ciphertext is valid and the adversary will output $c' = c$ with advantage $1/2 + \epsilon$; otherwise, T is universally random and the adversary's advantage is $1/2$. The simulation time is mainly dominated by the token simulation, and each token requires $O(n)$ exponentiations. The simulation is successful when $\mathbf{G}^* = \mathbf{G}_{i^*}$ which holds with $1/q_k$ probability. We therefore obtain the Theorem 1. □

Appendix C: Proof of Theorem 4

Proof. Suppose there exists an adversary who can break the membership encryption under maximum number accountability model. We construct an algorithm $\mathcal{B}$ that solves the (f, n)-DHE problem. $\mathcal{B}$ interacts with the adversary as the follows.

Challenge. Let $\mathbb{PG} = (\mathbb{G}, \mathbb{G}_T, e, p, g')$ be the pairing group. The algorithm $\mathcal{B}$ works as follows to simulate the system parameter.

- Randomly chooses β_0, γ_0 from $\mathbb{Z}_p$ and sets

  $$\alpha = a, \quad \beta = \beta_0, \quad \gamma = \gamma_0$$

  where a is the randomness in the challenge instance.
- $\mathcal{B}$ picks a random $y \in \mathbb{Z}_p$ and sets

  $$g = g_0, \quad h = g_0^y.$$

  We have

  $$e(g^\gamma, h) = e(g_0, g_0)^{y\gamma_0},\ \ g^{\alpha\gamma} = g_0^{a\gamma_0}, u_i = h^{\gamma\alpha^i} = (g_0^{a^i})^{y\gamma_0}, v_i = h^{\gamma\beta\alpha^i} = (g_0^{a^i})^{y\beta_0}.$$

– For $i = 1, 2, \cdots, n$, randomly choose s'_i from $\mathbb{Z}_p$, and set

$$s_i = \frac{(s'_i a + 1)(a + A_i)}{A_i}.$$

We have

$$\begin{aligned} d_{A_i} &= \left(g^{\frac{s_i}{(\alpha+A_i)(\beta+A_i)}},\ h^{\frac{s_i-1}{\alpha}},\ h^{s_i},\ h^{s_i\alpha},\ \cdots,\ h^{s_i\alpha^{n-2}} \right) \\ &= \left(g_0^{\frac{s_i a+1}{A_i(\beta_0+A_i)}},\ g_0^{y(\frac{s'_i}{A_i}a+s'_i+\frac{1}{A_i})},\ g_0^{s_i y},\ \cdots, g_0^{s_i a^{n-2} y} \right). \end{aligned}$$

All elements are computable from the challenge instance and setting. $\mathcal{B}$ generates the system parameter and sends it to the adversary.

Win: The adversary outputs $(\mathcal{P}(\mathbf{G}), \mathbf{G}, \mathbf{S})$, where $\mathbf{G} = \{A_1, A_2, \cdots, A_k\}$ but the verification shows that $|\mathcal{P}(\mathbf{G})| < k$. Let $\mathcal{P}(\mathbf{G}) = (w_1, w_2, w_3)$. If the verification shows $|\mathcal{P}(\mathbf{G})| = k' < k$, we can write $\mathcal{P}(\mathbf{G})$ into

$$\mathcal{P}(\mathbf{G}) = (w_1, w_2, w_3) = \left(w_1,\ \ w_1^{\beta},\ \ w_1^{\beta\alpha^{n-k'}} \right).$$

Let $\mathbf{S} = \tau$, we have

$$w_1 = h^{\tau \prod_{i=1}^{k}(\alpha+A_i)} = g_0^{y\tau \prod_{i=1}^{k}(a+A_i)}.$$

The algorithm $\mathcal{B}$ sets $f(x) = y\beta\tau x^{n-k'} \prod_{i=1}^{k}(x + A_i)$, which is an $(n + k - k')$-degree polynomial function in $\mathbb{Z}_p[x]$ and outputs $(f(x), w_3)$ as the solution to the (f, n)-DHE problem.

This completes the description of our simulation and we obtain the Theorem 4. □